Naive Bayes

Conditional Probability:

Consider two events A and B. Assume $P[B] \neq 0$
The conditional probability of A given B is

$$P[A|B] = \frac{P[A \cap B]}{P[B]}$$

It is the probability that A happens when we know that B has already happened.

- It is a measure. It measures the relative size of A inside B.

Independence:

Two events A and B are statistically independent if

$$P[A \cap B] = P[A] \cdot P[B]$$

why define independence in this way?

Recall that $P[A|B] = \dfrac{P[A\cap B]}{P[B]}$.

If A and B are independent, then $P[A\cap B] = P[A]P[B]$ and so

$$P[A|B] = \frac{P[A\cap B]}{P[B]} = \frac{P[A]P[\cancel{B}]}{P[\cancel{B}]} = P[A]$$

This suggest and interpretation of independence:

If the occurrence of B provides no additional information about the occurrence of A, then A and B are independent.

Therefore, we can define independence via conditional probability:

$$\boxed{\begin{array}{l}\text{Let A and B be two events such that } P[A] > 0 \text{ and } P[B] > 0. \\ \text{Then,} \\ \quad \text{A and B are independent if} \\ \quad\quad P[A|B] = P[A] \text{ or } P[B|A] = P[B].\end{array}}$$

This is because $P[A|B] = P[A\cap B]/P[B]$.
If $P[A|B] = P[A]$ then $P[A\cap B] = P[A]P[B]$,

which implies that $P[B|A] = P[A \cap B] / P[A] = P[B]$.

Bayes' Theorem

Bayes' Theorem: For any two events A and B such that $P[A] > 0$ and $P[B] > 0$,

$$P[A|B] = \frac{P[B|A]P[A]}{P[B]}$$

Proof:

By the definition of conditional probabilities, we have

$$P[A|B] = \frac{P[A \cap B]}{P[B]} \quad \text{and} \quad P[B|A] = \frac{P[B \cap A]}{P[A]}$$

Rearranging the terms yields

$$P[A|B]P[B] = P[B|A]P[A]$$

$$\because P[A \cap B] = P[B \cap A]$$

$$\therefore \quad P[A|B] = \frac{P[B|A]P[A]}{P[B]}$$

Bayes' theorem provides two views of the intersection $P[A \cap B]$ using two different conditional probabilities.

we call $P[B|A] \rightarrow$ conditional probability and $P[A|B] \rightarrow$ posterior probability

The order of A and B is arbitary. we can also call $P[A|B]$ the conditional probability and $P[B|A]$ the posterior probability. The context of the problem will make this clear.

when do we need to use Bayes' theorem?

Bayes' theorem switches the role of the conditioning, from $P[A|B]$ to $P[B|A]$.

Example:

P[win the game | play with A] and
P[play with A | win the game]

$P[W|A] \rightarrow$ conditional probability / likelihood P.
$P[A] \rightarrow$ prior probability
$P[A|W] \rightarrow$ posterior probability

Naive Bayes Classifier

Dataset Format:

$$x = \{x_1, x_2, x_3, \ldots\ldots, x_n\} \{y\}$$

dependent Features

Independent Features

Bayes theorem:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Now we have to find probability of y given that all the features have already occured:

$$P(y|x_1, x_2, \ldots, x_n) = \frac{P(x_1|y) \cdot P(x_2|y) \ldots P(x_n|y) \times P(y)}{P(x_1) \cdot P(x_2) \ldots P(x_n)}$$

$$= \frac{P(y) \cdot \prod_{i=1}^{n} P(x_i|y)}{P(x_1) \cdot P(x_2) \ldots P(x_n)}$$

Now, we can consider the denominator to be constant because this denominator will be same for every records.

$\therefore P(x_1) \cdot P(x_2) \cdots P(x_n) = \text{constant}$

$$\therefore P(y|x_1, x_2, \ldots, x_n) \propto P(y) \cdot \prod_{i=1}^{n} P(x_i|y)$$

$$y = \text{argmax}_y \; P(y) \cdot \prod_{i=1}^{n} P(x_i|y)$$

→ argmax means in $P(y)$ and $\prod_{i=1}^{n} P(x_i|y)$ which one gives us the highest probability, we will consider that.

Example to understand Naive Bayes Classifier

Here we have two features outlook and temperature and the problem statement is we have to predict whether the person is going to play tennis or not.

Outlook

	Yes	No	P(Y)	P(N)
Sunny	2	3	2/9	3/5
overcast	4	0	4/9	0
Rainy	3	2	3/9	2/5
Total →	9	5	100%	100%

temperature

	Yes	No	P(Y)	P(N)
Hot	2	2	2/9	2/5
mild	4	2	4/9	2/5
cool	3	1	3/9	1/5
Total →	9	5	100%	100%

Play

		P(Y) & P(N)
Yes	9	9/14
No	5	5/14
Total	14	100%

In outlook feature we have 3 category :-
Sunny Overcast Rainy

In Temperature feature we have 3 category
Hot, mild, cool

Now, we have to predict if the outlook is sunny and temperature is Hot, the person will go to play tennis or not using Naive Bayes.

Today(Sunny, Hot)

$$P(Yes/Today) = \frac{P(Sunny/Yes) \cdot P(Hot/Yes) \cdot P(Yes)}{P(today)}$$

$$\therefore P(Yes/today) = \frac{2}{9} \times \frac{2}{9} \times \frac{9}{14} = 0.031$$

we will skip P(today) because for all records this value is going to be same.

$$\therefore P(No/Today) = \frac{P(Sunny/No) \cdot P(Hot/No) \cdot P(No)}{P(today)}$$

$$= \frac{3}{5} \times \frac{3}{5} \times \frac{5}{14} = \underline{0.0857}$$

Now to calculate the probability of Yes with respect to today condition.

we have to normalize P(Yes)
so, to normalized

$$P(\text{yes}) = \frac{0.031}{0.031 + 0.0857}$$

$$\approx \underline{\underline{0.27}}$$

Similarly Normalize P(NO)

$$P(NO) = 1 - 0.27 \approx \underline{\underline{0.73}}$$

Now, here we see that probability of No is greater than probability of Yes.

So, in this condition when outcome is sunny and temperature is hot the person will not go to play tennis.

Naive Bayes' Classifier on text data

Text classification are like Spam or Ham.
Good are Bad.

Naive Bayes is considered as base line algorithm for text classification.

Example 1:

sentence 01: The food is Delicious.
sentence 02: The food is Bad.
sentence 03: Food is Bad.

Now based on the sentence predict whether the sentence is Good or Bad.

This problem is of NLP [Natural Language Processing]

In NLP, for this type problem first we need to do lot of preprocessing, which ~~means~~ includes

- ~~Remo~~ Removing stopwords
- perform stemming
- Bag of words (BOW)

→ TFIDF : Term frequency - Inverse document frequency

	f_1	f_2	f_3	f_4	
	The	food	Delicious	Bad	O/P
sentence 01	1	1	1	0	1
sentence 02	1	1	0	1	0
sentence 03	0	1	0	1	0
	0	1	1	0	1
	0	0	0	1	0

$$\text{sentence} = [x_1, x_2, x_3, \ldots, x_n]$$

sentence made of words $x_1, x_2, x_3, \ldots, x_n$

according to Bayes' Theorem :

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

$$P(y = \text{yes} \mid \text{sentence}) = P(y = \text{yes} \mid (x_1, x_2, \ldots x_n)) * P(y)$$

$$\therefore P(y=\text{yes} \mid \text{sentence}) \propto P(y) \times \prod_{i=1}^{n} P(x_1 \mid y=\text{yes}) \times P(x_2 \mid y=\text{yes}) \ldots \times P(x_n \mid y=\text{yes})$$

Now from Bag of words we found out the most frequent words.

"The" "Food" "Delicious" "Bad"

From the sentence " The Food is Delicious"

- " The" = x_1 "Delicious" = x_3
  "Food" = x_2 "Bad" = x_4

$P(y = yes) = \frac{2}{5}$ { as we have 2 yes and 3 No in the O/P feature }

$P(x_1 | y = yes) = \frac{1}{2}$ { as we have only one 1 in the O/P where "The" is present two times }

$P(x_2 | y = yes) = \frac{2}{4} = \frac{1}{2}$ { as we have 2 ones in the O/P where "Food" is present 4 times }

$P(x_3 | y = yes) = \frac{2}{2} = 1$

So comming back to our equation.

$$P(y=yes \mid sentence) \propto P(\overset{x_1 \mid y=yes}{\cancel{y=yes \mid x_1}}) * P(x_2 \mid y=yes) * P(x_3 \mid y=yes) * P(y=yes)$$

$$\therefore \quad P(y=yes \mid sentence) = \frac{1}{2} \times \frac{1}{2} \times 1 \times \frac{2}{5}$$

$$= \frac{1}{10}$$

$$P(y=yes \mid sentence) = 0.1$$

Similarly, now we have to calculate peobability of $(y=No \mid sentence)$

$$\therefore \; P(y=No \mid sentence) = P(y=No) \cdot P(x_1 \mid y=No) \cdot P(x_2 \mid y=No) \cdot P(x_4 \mid y=No)$$

$$= \frac{3}{5} \times \frac{1}{2} \times \frac{2}{4} \times \frac{3}{3}$$

$$= \frac{3}{20} = 0.15$$

$$P(y=No \mid sentence) = 0.15$$

Now we will normalize both P(y=yes| sentence) and P(y=No| sentence)

∴ after normalize

$$P(y=yes|sentence) = \frac{0.1}{0.1+0.15} = 0.25$$

$$\therefore P(y=no|sentence) = 1-0.25 = 0.75$$

In both P(yes) or P(No), which ever will have the highest probability, that will be considered as the O/P of that particular sentence.

For all the text classification in ML, it is always better to go with Naive Bayes, because in Naive Bayes the probability of each and every word will be checked, with respect to the feature that we have.

where does Naive Baye's Fail in case of Test Data?

Suppose in the sentence "The food is Delicious". we have a new word like " The food is Delicious Tasty".

Now Tasty is the word that is not present in our features, and therefore when we find the probability of that sentence the value will be zero.

∴ it will be treated as negative output.